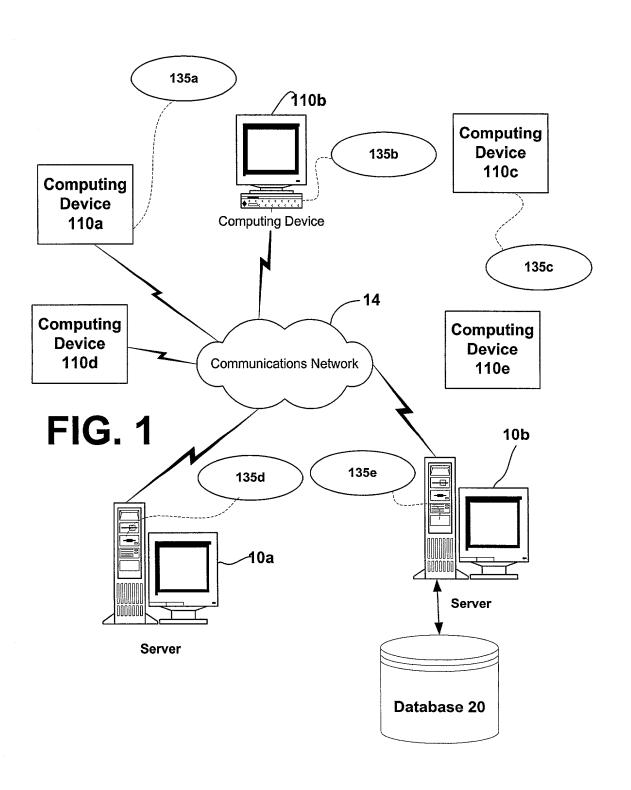Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 1 of 15

135a

110b

135b

Computing
Device
110c

Computing
Device
110a

Computing Device

Computing
Device
110d

135c

14

Communications Network

Computing
Device
110e

FIG. 1

135d

135e

10b

10a

Server

Server

Database 20

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
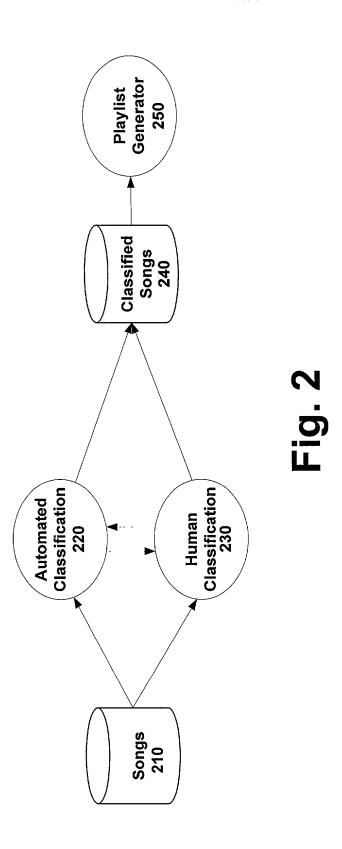Title: System and Methods for Providing Adaptive
Property Classification
Page 2 of 15

Fig. 2

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 3 of 15

Fig. 3

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
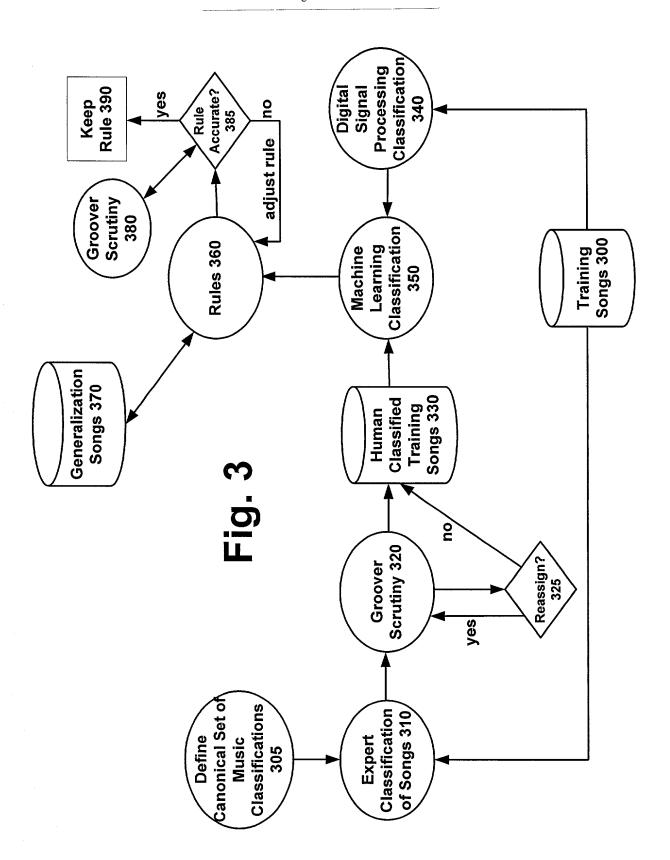Property Classification
Page 4 of 15

Fig. 4A

ME ? +

CH11, CH12, CH13, ... , CH1N
CH21, CH22, CH23, ... , CH2N
$\vdots$
CHM1, CHM2, CHM3, ... ,CHMN

+ Expert = ME {

CH13
CH21
$\vdots$
CHM7

Fig. 4B

ME ? +

CC11, CC12, CC13, ... , CC1N
CC21, CC22, CC23, ... , CC2N
$\vdots$
CCM1, CCM2, CCM3, ... ,CCMN

+ DSP = ME {

CC11
CC25
$\vdots$
CCM2

Fig. 4C

ME =

CH13, CH21, ..., CHM7, CC11, CC25, ..., CCM2 →

Vector V_{ME}

Fig. 4D

Neighborhood

d_nh ─── EV1

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
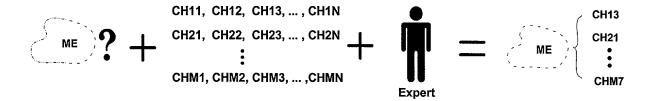Property Classification
Page 5 of 15

Fig. 5A

Classification Chain Input Space IS

Fig. 5B

error

e4

e3

e2
e1

d6   d1   d4

d_nh

distance of nearest
neighbor for each of
EV1, EV2 and EV3

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 6 of 15

Fig. 6A

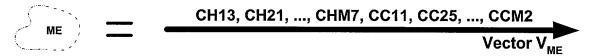**Classification Chain**
Input Space IS

Fig. 6B

Classification Chain
Input Space IS

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 7 of 15

600
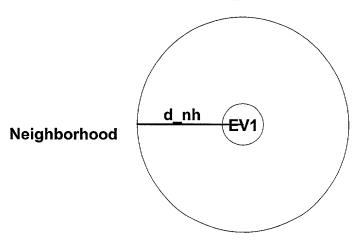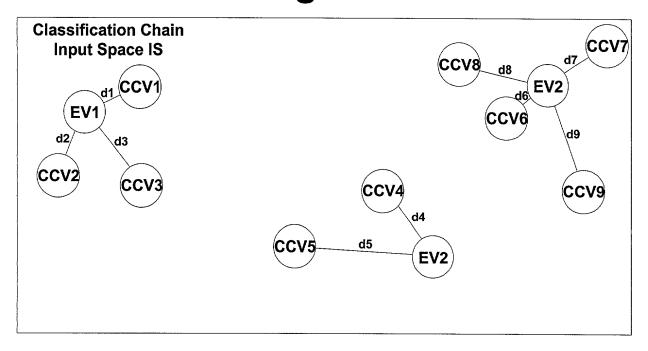Start with the First Song of a Plurality of Unclassified Songs

605
Is Song Already in Database?
YES →
NO ↓

610
Is Training Complete?
NO →
YES ↓

615
Next Song of Plurality of Unclassified Songs

620
Calculate and Present Song Vector to Classification Chain

625
Quit

630
Calculate Response of Classification Chain

Fig. 6C

635
Is there a High Level of Confidence for Chain Output?
NO →
YES ↓

640
Expert Classifies Song, and Song Vector Added to Classification Chain

645
Next Song of Plurality of Unclassified Songs

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
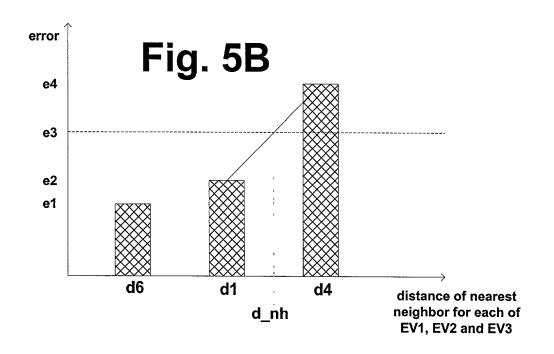Title: System and Methods for Providing Adaptive
Property Classification
Page 8 of 15

Fig. 7A

745
FFT

750
Critical Band Filtering

755
Entropy

760
Derivative

765
Average of All Frames

770
Feature Vector

775
Principal Component Analysis

780
Classification Chain

Fig. 7B

High

Sad ← → Happy

Emotion

Low

Mood

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 9 of 15

Fig. 8A

800

FFT

802

Peak Detection

804

Critical Band Filtering

806

Peak Continuation

808

Consonance Calculation

810

Feature
Vector

812

Principal
Component
Analysis

814

Classification
Chain

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
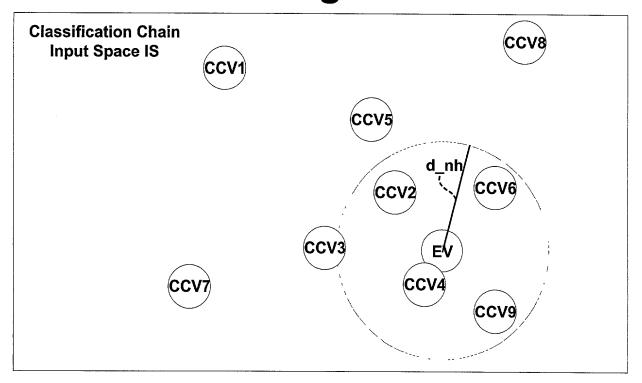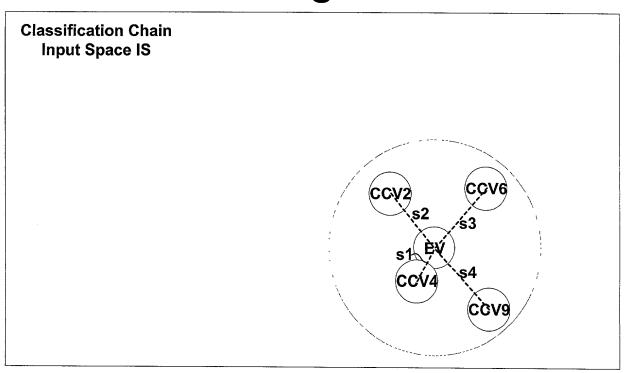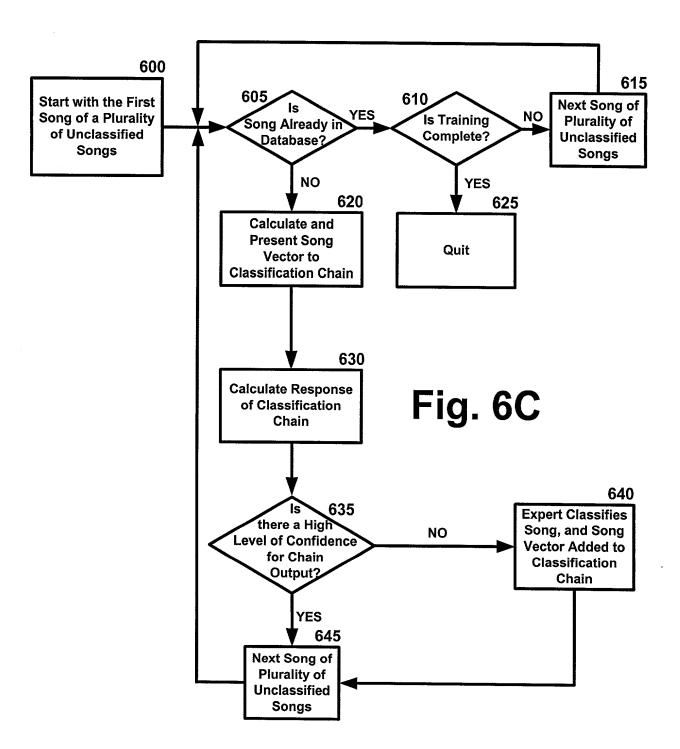Property Classification
Page 10 of 15

Fig. 8B

816
FrameNum = 1 : NumFrames

818
Zero-pad and Take FFT of Audio Frame

820
Diff_FFT[ ] = Successive Differences of FFT'd Buffer

822
FFTIndex = 1:NumBins

824
Diff_FFT(FFTIndex) > 0? —NO→

↓YES

826
Diff_FFT(FFTIndex+1) < 0? —NO→

↓YES

828
Record Bin & Energy into Output Matrices

830
FFTIndex++ FFTIndex >= LastBin?
NO
YES

834
Select Top Requested Number of Peaks

836
PeakNum = 1: NumPeaksRequested

838
Nth Order Interpolate Peaks' Positions and Heights

840
PeakNum++ PeakNum > NumPeaksRequested? —NO→

↓YES

842
FrameNum++ FrameNum > NumFrames? —NO→

↓YES

844
FINISHED

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
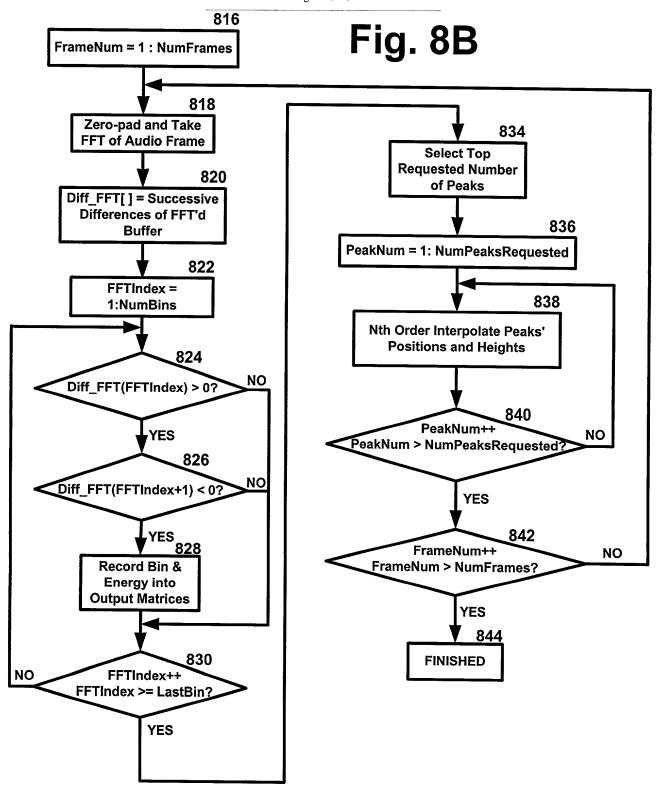Property Classification
Page 11 of 15

846
Setup Guides Based on Peaks
in First Frame

848
For FrameNum = 1: NumFrames

850
PeakMatched = [ ]
GuideMatch = 0

852
For PeakNum = 1:NumPeaks

854
MinDiff = kNyquistFreq
Matched = 0
ThisPeak = Bins(FrameNum, PeakNum)
ThisEnergy = Energies(FrameNum, PeakNum)

856
For GuideNum = 1:NumGuides

858
ThisGuide = GuideFreq(GuideNum)
ThisGuideEnergy = GuideEnergy(GuideNum)

860
If ((ThisPeak
Is within 1/2 Step of ThisGuide)
& (PeakEnergy is within 6dB of
ThisGuideEnergy) & (ThisGuide
not Just Continued))

NO

YES

862
GuideMatch = GuideNum
(Record Matched Guide #)

864
guideNum++
(guideNum>numGuides)?

NO

YES

866
GuideMatch != 0

NO

YES

868
- Record this Peak & Energy into Output
Matrix
- Record Possible Missed Peak in
Previous Frame if ThisGuide Just Started
- PeakMatched(PeakNum) = 1
- GuideState(GuideNum) = kContinued
- GuideEnergy(GuideNum) = ThisEnergy

870
PeakNum++
PeakNum > NumPeaks?

NO

YES

872
- Locate & Record Unused Guides
and Unmatched Peaks

874
Any Unused Guides
or Unmatched Peaks?

NO

YES

876
Start New Guides at the
Unmatched Peaks with Largest
Energy

878
FrameNum++
FrameNum > NumFrames?

NO

YES

879
FINISHED

Fig. 8C

Fig. 8D

880
FrameNum = 1 : NumFrames

882
PeakBottom = 1 : NumPeaks

884
Denom = Peak_Locations(FrameNum, PeakBottom)

886
Denom != 0? — NO

YES

888
PeakTop = 1 : NumPeaks

890
Numer = Peak_Locations(FrameNum, peakTop)

892
Numer != 0? — NO

YES

894
Ratio = Nearest_Integer [12*
log2 (Numer/Denom)]

896
Is (0<Ratio<25) — NO

YES

898
Increment Intervals(Ratio) Array by
(Energies(FrameNum, PeakBottom))^2 and
(Energies(FrameNum, PeakTop))^2

900
PeakTop++
PeakTop > NumPeaks? — NO

YES

902
PeakBottom++
PeakBottom > NumPeaks? — NO

YES

904
FrameNum++
FrameNum > NumFrames? — NO

YES

906
Interval = Interval / NumFrames
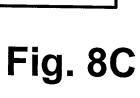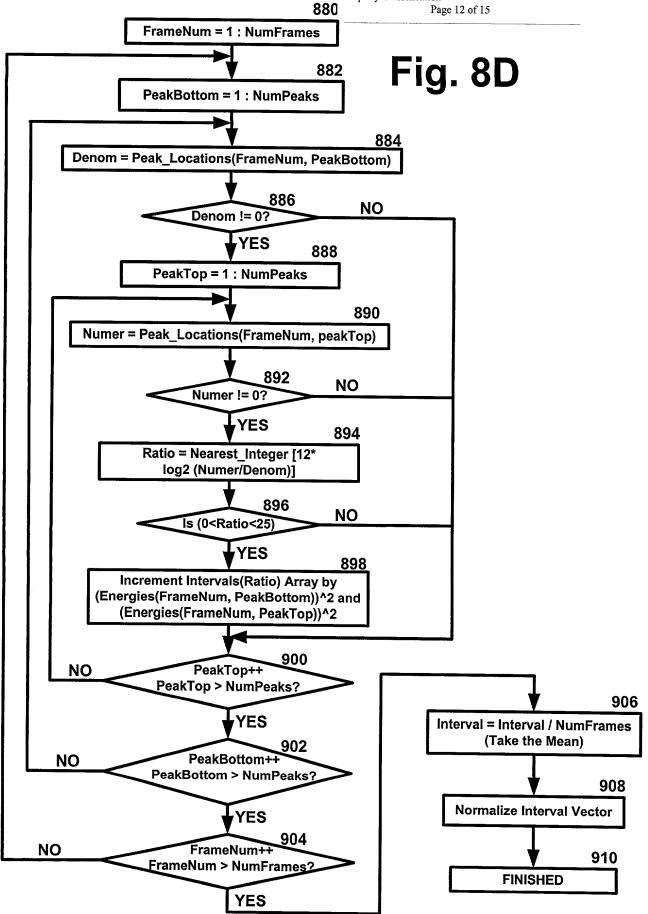(Take the Mean)

908
Normalize Interval Vector

910
FINISHED

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 13 of 15

Fig. 9A

```
                           900
        ┌──────────────────────┐
        │         FFT          │
        └──────────────────────┘
                    │
                    ▼       905
        ┌──────────────────────┐
        │    Peak Detection    │
        └──────────────────────┘
                    │
                    ▼       910
        ┌──────────────────────┐
        │ Critical Band Filtering │
        └──────────────────────┘
                    │
                    ▼       915
        ┌──────────────────────┐
        │   Peak Continuation  │
        └──────────────────────┘
                    │
                    ▼       920
        ┌──────────────────────┐
        │  Melodic Movement    │
        │     Calculation      │
        └──────────────────────┘
                    │
                    ▼    925
            ┌──────────────┐
            │   Feature    │
            │    Vector    │
            └──────────────┘
                    │
                    ▼       930
        ┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
            Principal
        │   Component       │
            Analysis
        └ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┘
                    │
                    ▼       935
            ┌──────────────┐
            │ Classification │
            │    Chain     │
            └──────────────┘
```

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 14 of 15

940

FrameNum = 1 : NumFrames

945

PeakNum = 1:numPeaks

950

Numer = Peak_Locations(FrameNum, PeakNum)

955

If (kFreqMin <= Numer <= kFreqMax)   NO

960

PitchClass = round(24*log2(Numer/kFreqMin) - floor(log2(Numer/kFreqMin))

Fig. 9B

965

Is (0<PitchClass<25)   NO

YES

970

Melodic(FrameNum, PitchClass) = Melodic(FrameNum, PitchClass) + (Energies(FrameNum, PeakNum))

YES

975

PeakNum++
PeakNum > NumPeaks?   NO

980

FrameNum++
FrameNum > NumFrames?   NO

985

Melodic = Diff(Melodic)
(Take the 1st Difference)

990

Melodic = std(Melodic)
(Take Standard Deviation)

995

FINISHED

Docket No.: MSFT-0579/167505.2
Inventors: Christopher B. Weare
Filing Date: August 20, 2001
Serial No.: Not yet assigned
Title: System and Methods for Providing Adaptive
Property Classification
Page 15 of 15

Fig. 10

Audio data in 16bit 44khz mono PCM format. 1020

1030
Decimate data to common sampling rate, e.g. 43 Hz.

1040
Generate a data envelope that accentuates tempo characteristics while suppressing biases and trends.

1050
Generate 130 bins of data, where each bin is the "correlation strength" for the period defined by the bin number times 1/43.

1060
Feed into classification chain. For every calculated distance that is below a defined threshold, the corresponding classified tempo along with the distance is added to a table. The distance values in the table are translated into confidence values by taking 1/(distance^4). Each tempo value is divided by 10 and the log2 is taken. The integer portion is the "harmonic component" entry. The fractional portion is the "tempo component" entry.

1070
Tempo components are translated into vector representation. The range of 0 to 1 is mapped to 0 to 2pi. The sine and cosine of the angle are taken and multiplied by the corresponding confidence value. These sine and cosine components for the entire table are accumulated separately to create an overall representative vector. The angle of this vector is mapped back to a range of 0 to 1 to give an overall confidence of the tempo classification.

1080
The harmonic component of each table entry is evaluated. If the corresponding tempo component meets one of the following criteria, the harmonic is modified. If the tempo entry is less than the overall tempo classification - 0.5, the harmonic component entry is decremented. If the tempo entry is greater than the overall tempo classification + 0.5, the harmonic component is incremented. Each harmonic component table entry "votes" for its value with a weight proportional to the corresponding confidence value. The most prominent value is used as the overall harmonic component.

1090
The overall tempo is calculated by calculating 2^(overall harmonic component + overall tempo component) multiplied by 10.